

EUR 2241.f

LIBRARY COPY

COMMUNAUTE EUROPEENNE DE L'ENERGIE ATOMIQUE - EURATOM

CONTOUR MAP SUBROUTINES

par

P. MOINIL et J. PIRE

1965



Centre Commun de Recherche Nucléaire
Etablissement Ispra - Italie

Centre de traitement de l'information scientifique - CETIS

AVERTISSEMENT

Le présent document a été élaboré sous les auspices de la Commission de la Communauté Européenne de l'Energie Atomique (EURATOM).

Il est précisé que la Commission d'EURATOM, ses cocontractants ou toute personne agissant en leur nom :

- 1° — Ne garantissent pas l'exactitude ou le caractère complet des informations contenues dans ce document, ni que l'utilisation d'une information, d'un équipement, d'une méthode ou d'un procédé décrit dans le présent document ne portent pas atteinte à des droits privatifs.
- 2° — N'assument aucune responsabilité pour les dommages qui pourraient résulter de l'utilisation d'informations, d'équipements, de méthodes ou procédés divulgués dans le présent document.

Ce rapport est vendu au prix de 40,— francs belges, sur demande adressée à : PRESSES ACADEMIQUES EUROPEENNES — 98, Chaussée de Charleroi, Bruxelles 6.

Le paiement se fait par versement à la :

- BANQUE DE LA SOCIETE GENERALE (Agence Ma Campagne) - Bruxelles - compte N° 964.558,
- BELGIAN AMERICAN BANK AND TRUST COMPANY - New York - compte N° 22.186,
- LLOYDS BANK (Europe) Ltd. - 10, Moorgate, London E.C.2,

en mentionnant la référence : « EUR 2241.f CONTOUR MAP SUBROUTINES »

Le présent document a été reproduit à partir de la meilleure copie disponible.

EUR 2241.f

COMMUNAUTE EUROPEENNE DE L'ENERGIE ATOMIQUE - EURATOM

CONTOUR MAP SUBROUTINES

par

P. MOINIL et J. PIRE

1965



Centre Commun de Recherche Nucléaire
Etablissement Ispra - Italie

Centre de traitement de l'information scientifique - CETIS

Manuscrit reçu le 10-12-1964.

CONTOUR MAP SUBROUTINES

La représentation par "lignes de niveau" est fréquemment utilisée pour illustrer le comportement d'une fonction de deux variables. Jusqu'à présent ce genre de travail devait être exécuté soit à la main, soit à l'aide de machines spécialisées.

La subroutine COMAP (Contour Map) produit les informations nécessaires pour conduire automatiquement un Data Plotter Calcomp dans le tracé de tels dessins.

COMAP est écrit en Fortran II; outre les sousroutines habituelles du système, certaines routines d'usage courant pour l'utilisation du Data Plotter (PL/T, PLTIR, NUMBER, SYMBL4: voir réf. 2), elle utilise une routine spéciale à 3 entrées (CLEAN, PLACE, INAL), écrite en FAP, pour tenir sa comptabilité des points utilisés.

COMAP est programmé de façon à fournir le dessin des lignes de niveau sous sa forme traditionnelle. Les valeurs de la fonction $z(x,y)$ sont rangées dans une matrice et sont calculées pour un certain incrément de x et de y .

Les valeurs pour un x constant constituent une ligne de la matrice; celles pour un y constant forment une colonne. L'incrément de x et celui de y peuvent être différents. Lors de la représentation on attribuera à l'incrément en x et à celui en y une certaine longueur (en cms). De plus, l'angle entre les axes x et y peut être choisi à la convenance de l'utilisateur et sera spécifié en degrés.

La matrice utilisée ne doit pas nécessairement être remplie; il est possible de spécifier la partie de matrice à traiter pour effectuer le dessin.

Les contours sont calculés par interpolation entre les points fournis dans la matrice, puis par une transformation linéaire, ajustés aux nécessités du dessin.

Les valeurs ne sont pas altérées par la subroutine.

La méthode d'interpolation et la logique utilisées sont celles décrites dans la référence 1; disons seulement qu'elles assurent partout une représentation satisfaisante de la fonction, aussi bien aux environs des maxima et minima qu'ailleurs.

En ce qui concerne la programmation nous avons suivi dans les lignes générales la subroutine de M. O. Dayhoff (réf. 1); nous avons cependant réduit l'occupation mémoire à moins de 30% de la version citée et levé toutes les restrictions au sujet de la longueur des contours. L'unique restriction est dans la dimension totale de la matrice (10.008); cependant cette occupation totale peut être reportée selon les nécessités du programme appelant. Au cas où cette valeur serait insuffisante pour un problème donné, il suffit de recompiler EUCAP (dont les entrées sont CLEAN, PLACE et INAL) en modifiant la longueur de la réservation REC (actuellement 278). Chaque mot ajouté à REC augmente de 36 mots la longueur totale possible de la matrice à traiter.

Utilisation de COMAP

La séquence d'appel de COMAP est la suivante:

CALL COMAP(AM,NXL,NYL,KX,KY,DELTAX,DELTAY,CNLEV,THETA,ID)

où : AM est la matrice contenant la fonction à représenter.

NXL est le nombre de points selon l'axe des x (y constant)

NYL est le nombre de points selon l'axe des y (x constant)

KX est la dimension de définition de AM pour un premier indice.

KY est la dimension de définition de AM pour un deuxième indice.

DELTAX est la longueur, exprimée en centimètres, de l'intervalle séparant la représentation de deux valeurs consécutives de la fonction pour une même valeur de y (il peut être positif ou négatif).

DELTAY est la longueur, exprimée en centimètres, de l'intervalle séparant la représentation de deux valeurs consécutives de la fonction pour une même valeur de x (il peut être négatif ou positif).

CNLEV est la valeur de la fonction pour laquelle on désire tracer le contour.

THETA est l'angle, exprimé en degrés, entre la direction positive de l'axe des x et la direction positive de l'axe des y.

ID est un indicateur. S'il vaut zéro, la ligne de niveau est tracée en trait plein, s'il vaut un, elle est tracée en tireté.

Avant d'utiliser COMAP, le programmeur doit conduire la plume du Data

Plotter à l'origine choisie (à l'aide d'un appel FINIM); les points correspondant aux valeurs fournies par la matrice auront par rapport à cette origine, les coordonnées

$(XI \pm DELTAX + (YJ \pm DELTAY) \pm \cos(THETA)), YJ \pm DELTAY \pm \sin(THETA).$

Pour un même THETA, on peut donc utiliser au choix une des 4 possibilités de combinaison de KX et KY, positifs et négatifs (voir exemple).

Un cadre peut être tracé en appelant la subroutine AXMAP:

CALL AXMAP(NXL,NYL,DELTAX,DELTAY,THETA,BCDX,NØX,BCDY,NØY)

où les 5 premiers arguments ont la même signification que dans COMAP;

BCDX est un vecteur contenant le titre à donner à l'axe des x,

NØX est le nombre de caractères de ce titre,

BCDY est un vecteur contenant le titre à donner à l'axe des y,

NØY est le nombre de caractères de ce titre.

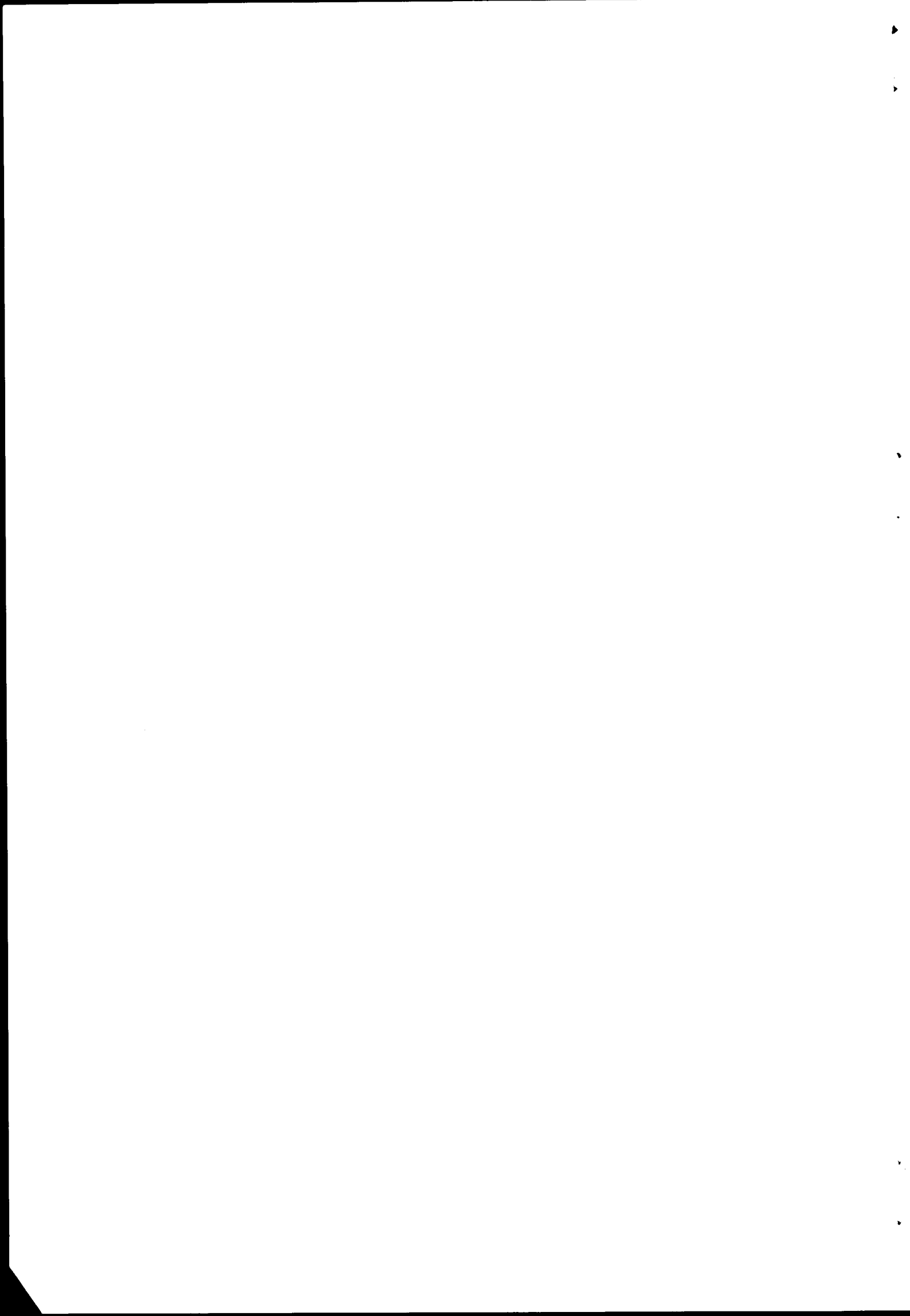
Remarquons que BCDX et BCDY doivent être définis par une expression du type $nHA_1A_2A_3\dots A_n$; voir SYMBL4 (réf. 2).

Occupation mémoire :

EUCHS (COMAP)	1255
EUCAP (CLEAN,PLACE,INAL)	316
EUAMS (AXMAP)	497
	<hr/>
	2068

Références :

1. A CONTOUR MAP PROGRAM for X-ray Crystallography.
M.O. DAYHOFF, National Biomedical Research Fondation,
Silver Spring, Maryland
(Communications of the Association for Computing Machinery,
Volume 6, Number 10, October 1963).
2. Programmation relative au Calcomp (rapport EUR en impression).




```

C      (7090 SUBROUTINE FOR CALCOMP 564/1401)
C*****
C* CALLING SEQUENCE
C* -----
C* 1. FORTRAN = CALL COMAP(AM,NXL,NYL,NXD,NYD,DELTAX,DELTAY,CL,
C*                      THETA,ID)
C* 2. FAP      = TSX      $COMAP,4
C*                PZE      AM
C*                PZE      NXL
C*                :
C*                :
C*                PZE      ID
C*
C*  AM = MATRICE CONTENANT LA VALEUR DES POINTS.
C*  NXL = DIMENSION UTILISEE DES X DANS LA MATRICE.
C*  NYL = DIMENSION UTILISEE DES Y DANS LA MATRICE.
C*  NXD = DIMENSION DEFINIE POUR LES X DANS LA MATRICE.
C*  NYD = DIMENSION DEFINIE POUR LES Y DANS LA MATRICE.
C*  DELTAX = INTERVALLE DE DESSIN POUR LES X.
C*  DELTAY = INTERVALLE DE DESSIN POUR LES Y.
C*  CL = VALEUR DE LA COURBE DE NIVEAU DESIREE.
C*  THETA = ANGLE EN DEGRES(AXE Y).
C*  ID = 0 POUR TRAIT PLEIN, 1 POUR TIRETS.
C*****
C      ( J. PIRE - 1/11/64 )
C
C      C.C.R.EURATOM
C      (ISPRA)
C
C      SUBROUTINE COMAP(AM,NXL,NYL,NXD,NYD,DELTAX,DELTAY,CL,THETA,ID)
C
C      DIMENSION AM(1)
C      DIMENSION IPT(3,3),INX(8),INY(8)
C
C      XNDEXF(IXRX,IXRY) = MD * (IXRY - 1) + IXRX
C      PTXF(PX,PY) = (PX * PASX) + (PY * RC)
C      PTYF(PY) = PY * RS
C
C      MT = NXL
C      NT = NYL
C      MD = NXD
C      ND = NYD
C      PASX = DELTAX
C      PASY = DELTAY
C      CV = CL
C      LD = ID
C      THE = THETA * 0.017453294
C      RC = COSF(THE) * PASY
C      RS = SINF(THE) * PASY
C      IF (IZX) 102,101,102
101 IPT(1,1) = 8
    IPT(1,2) = 1

```

```

CMS00010
CMS00015
CMS00020
CMS00030
CMS00040
CMS00050
CMS00060
CMS00070
CMS00080
CMS00090
CMS00100
CMS00110
CMS00120
CMS00130
CMS00140
CMS00150
CMS00160
CMS00170
CMS00180
CMS00190
CMS00200
CMS00210
CMS00220
CMS00230
CMS00240
CMS00250
CMS00260
CMS00270
CMS00280
CMS00290
CMS00300
CMS00310
CMS00320
CMS00330
CMS00340
CMS00350
CMS00360
CMS00370
CMS00380
CMS00390
CMS00400
CMS00410
CMS00420
CMS00430
CMS00440
CMS00450
CMS00460
CMS00470
CMS00480
CMS00490
CMS00500
CMS00510
CMS00520
CMS00530
CMS00540

```

```

      IPT(1,3) = 2
      IPT(2,1) = 7
      IPT(2,3) = 3
      IPT(3,1) = 6
      IPT(3,2) = 5
      IPT(3,3) = 4
      INX(1) = -1
      INX(2) = -1
      INX(3) = 0
      INX(4) = 1
      INX(5) = 1
      INX(6) = 1
      INX(7) = 0
      INX(8) = -1
      INY(1) = 0
      INY(2) = 1
      INY(3) = 1
      INY(4) = 1
      INY(5) = 0
      INY(6) = -1
      INY(7) = -1
      INY(8) = -1
      IZX = 2
102  CALL CLEAN
      ISS = 0
      MT1 = MT - 1
      I = 1
108  IF (AM(I) - CV) 103,105,105
103  IF (AM(I+1) - CV) 105,104,104
104  IX = I + 1
      IY = 1
      IDX = -1
      IDY = 0
      ASSIGN 105 TO NRET
      GO TO 300
105  IF (I - MT1) 106,107,106
106  I = I + 1
      GO TO 108
107  NT1 = NT - 1
      I = 1
114  INDEX1 = XNDEXF(MT,I)
      IF (AM(INDEX1) - CV) 109,110,110
109  INDEX1 = XNDEXF(MT,I+1)
      IF (AM(INDEX1) - CV) 110,111,111
111  IX = MT
      IY = I + 1
      IDX = 0
      IDY = -1
      ASSIGN 110 TO NRET
      GO TO 300
110  IF (I - NT1) 112,113,112
112  I = I + 1
      GO TO 114
113  I = 1
120  MT2 = MT + 1 - I
      INDEX1 = XNDEXF(MT2,NT)
```

```

CMS00550
CMS00560
CMS00570
CMS00580
CMS00590
CMS00600
CMS00610
CMS00620
CMS00630
CMS00640
CMS00650
CMS00660
CMS00670
CMS00680
CMS00690
CMS00700
CMS00710
CMS00720
CMS00730
CMS00740
CMS00750
CMS00760
CMS00770
CMS00780
CMS00790
CMS00800
CMS00810
CMS00820
CMS00830
CMS00840
CMS00850
CMS00860
CMS00870
CMS00880
CMS00890
CMS00900
CMS00910
CMS00920
CMS00930
CMS00940
CMS00950
CMS00960
CMS00970
CMS00980
CMS00990
CMS01000
CMS01010
CMS01020
CMS01030
CMS01040
CMS01050
CMS01060
CMS01070
CMS01080
CMS01090
CMS01100
```

```

      IF (AM(INDEX1) - CV) 115,116,116
115  INDEX1 = XNDEXF(MT2-1,NT)
      IF (AM(INDEX1) - CV) 116,117,117
117  IX = MT2 - 1
      IY = NT
      IDX = 1
      IDY = 0
      ASSIGN 116 TO NRET
      GO TO 300
116  IF (I - MT1) 118,119,118
118  I = I + 1
      GO TO 120
119  I = 1
126  NT2 = NT + 1 - I
      INDEX1 = XNDEXF(1,NT2)
      IF (AM(INDEX1) - CV) 121,122,122
121  INDEX1 = XNDEXF(1,NT2-1)
      IF (AM(INDEX1) - CV) 122,123,123
123  IX = 1
      IY = NT2 - 1
      IDX = 0
      IDY = 1
      ASSIGN 122 TO NRET
      GO TO 300
122  IF (I - NT1) 124,125,124
124  I = I + 1
      GO TO 126
125  ISS = 1
      J = 2
135  I = 1
129  INDEX1 = XNDEXF(I,J)
      IF (AM(INDEX1) - CV) 127,128,128
127  INDEX1 = XNDEXF(I+1,J)
      IF (AM(INDEX1) - CV) 128,130,130
130  IF (INALF(XNDEXF(I+1,J))) 131,128,131
131  IX = I + 1
      IY = J
      IDX = -1
      IDY = 0
      ASSIGN 128 TO NRET
      TRACE PROGRAM.
C
C
300  XPY = 0.
      FLAG = 0.
      JT = 0
      IX0 = IX
      IY0 = IY
      ISX = IDX + 2
      ISY = IDY + 2
      IS = IPT(ISX,ISY)
      ISO = IS
      IF (ISO - 8) 302,302,301
301  ISO = ISO - 8
302  IT = 0
323  ASSIGN 303 TO MRET

```

```

CMS01110
CMS01120
CMS01130
CMS01140
CMS01150
CMS01160
CMS01170
CMS01180
CMS01190
CMS01200
CMS01210
CMS01220
CMS01230
CMS01240
CMS01250
CMS01260
CMS01270
CMS01280
CMS01290
CMS01300
CMS01310
CMS01320
CMS01330
CMS01340
CMS01350
CMS01360
CMS01370
CMS01380
CMS01390
CMS01400
CMS01410
CMS01420
CMS01430
CMS01440
CMS01450
CMS01460
CMS01470
CMS01480
CMS01490
CMS01500
CMS01510
CMS01520
CMS01530
CMS01540
CMS01550
CMS01560
CMS01570
CMS01580
CMS01590
CMS01600
CMS01610
CMS01620
CMS01630
CMS01640
CMS01650
CMS01660

```

```
GO TO 500
303 IF (IT + JT - 1) 305,305,304
304 XS = X1
  YS = Y1
  X1 = X2
  Y1 = Y2
  X2 = XS
  Y2 = YS
305 IF (FLAG) 307,306,307
306 X = PTXF(X2,Y2)
  Y = PTYF(Y2)
  CALL NUMBER(X,Y,0.14,0.,CV,1)
  CALL PLOT(X,Y,3)
  FLAG = 2.
  X1 = X2
  Y1 = Y2
  GO TO 308
307 ASSICN 308 TO LRET
  GO TO 700
308 IS = IS + 1
  JT = IT
333 IF (IS - 9) 312,311,311
311 IS = IS - 8
312 IDX = INX(IS)
  IDY = INY(IS)
  IX1 = IX + IDX
  IY1 = IY + IDY
  IF (ISS) 317,317,313
313 IF (IX - IX0) 321,314,321
314 IF (IY - IY0) 321,315,321
315 IF (IS - ISO) 321,316,321
316 ASSIGN 309 TO MRET
  GO TO 500
309 ASSIGN 310 TO LRET
  GO TO 700
317 IF (IX1) 318,334,318
318 IF (IX1 - MT) 319,319,334
319 IF (IY1) 320,334,320
320 IF (IY1 - NT) 321,321,334
321 INDEX1 = XNDEXF(IX1,IY1)
  IF (CV - AM(INDEX1)) 322,322,323
322 IF (IDX**2 + IDY**2 - 1) 324,329,324
324 INDEX1 = XNDEXF(IX,IY)
  INDEX2 = XNDEXF(IX1,IY)
  INDEX3 = XNDEXF(IX,IY1)
  INDEX4 = XNDEXF(IX1,IY1)
  DCP = (AM(INDEX1) + AM(INDEX2) + AM(INDEX3) + AM(INDEX4)) / 4.0
  IF (DCP - CV) 323,325,325
325 IF (INX(IS-1)) 326,327,326
326 IX = IX + IDX
  IDY = -IDY
  XPY = 2.0
  ASSIGN 328 TO MRET
  GO TO 500
328 IX = IX + IDX
  ASSIGN 329 TO LRET
```

CMS01670
CMS01680
CMS01690
CMS01700
CMS01710
CMS01720
CMS01730
CMS01740
CMS01750
CMS01760
CMS01770
CMS01780
CMS01790
CMS01800
CMS01810
CMS01820
CMS01830
CMS01840
CMS01850
CMS01860
CMS01870
CMS01880
CMS01890
CMS01900
CMS01910
CMS01920
CMS01930
CMS01940
CMS01950
CMS01960
CMS01970
CMS01980
CMS01990
CMS02000
CMS02010
CMS02020
CMS02030
CMS02040
CMS02050
CMS02060
CMS02070
CMS02080
CMS02090
CMS02100
CMS02110
CMS02120
CMS02130
CMS02140
CMS02150
CMS02160
CMS02170
CMS02180
CMS02190
CMS02200
CMS02210
CMS02220

```
GO TO 700
327 IY = IY + IDY
    IDY = -IDY
    XPY = 2.0
    ASSIGN 330 TO MRET
C
C   CALCULATION PROGRAM.
C
500 IT = 0
    IF (IDX**2 + IDY**2 - 1) 502,501,502
501 IF (IDX) 504,503,504
503 X2 = IX
    Z = IY
    IY2 = IY + IDY
    DY = IDY
    INDEX1 = XNDEXF(IX,IY)
    INDEX2 = XNDEXF(IX,IY2)
    Y2 = ((AM(INDEX1) - CV) / (AM(INDEX1) - AM(INDEX2))) * DY + Z
    GO TO 550
504 Y2 = IY
    DX = IDX
    W = IX
    IX2 = IX + IDX
    INDEX1 = XNDEXF(IX,IY)
    INDEX2 = XNDEXF(IX2,IY)
    X2 = ((AM(INDEX1) - CV) / (AM(INDEX1) - AM(INDEX2))) * DX + W
    GO TO 550
502 IX2 = IX + IDX
    IY2 = IY + IDY
    W = IX
    Z = IY
    DX = IDX
    DY = IDY
    INDEX1 = XNDEXF(IX,IY)
    INDEX2 = XNDEXF(IX2,IY)
    INDEX3 = XNDEXF(IX,IY2)
    INDEX4 = XNDEXF(IX2,IY2)
    DCP = (AM(INDEX1) + AM(INDEX2) + AM(INDEX3) + AM(INDEX4)) / 4.0
    IF (XPY - 2.0) 505,506,505
505 IF (DCP - CV) 506,506,507
506 AL = AM(INDEX1) - DCP
    V = .5 * (AL + DCP - CV) / AL
    X2 = V * DX + W
    Y2 = V * DY + Z
    XPY = 0.0
    GO TO 550
507 IT = 1
    AL = AM(INDEX4) - DCP
    V = .5 * (AL + DCP - CV) / AL
    X2 = -V * DX + W + DX
    Y2 = -V * DY + Z + DY
550 GO TO MRET,(303,309,328,330)
C
C
330 IY = IY + IDY
    ASSIGN 329 TO LRET
```

```
CMS02230
CMS02240
CMS02250
CMS02260
CMS02270
CMS02280
CMS02290
CMS02300
CMS02310
CMS02320
CMS02330
CMS02340
CMS02350
CMS02360
CMS02370
CMS02380
CMS02390
CMS02400
CMS02410
CMS02420
CMS02430
CMS02440
CMS02450
CMS02460
CMS02470
CMS02480
CMS02490
CMS02500
CMS02510
CMS02520
CMS02530
CMS02540
CMS02550
CMS02560
CMS02570
CMS02580
CMS02590
CMS02600
CMS02610
CMS02620
CMS02630
CMS02640
CMS02650
CMS02660
CMS02670
CMS02680
CMS02690
CMS02700
CMS02710
CMS02720
CMS02730
CMS02740
CMS02750
CMS02760
CMS02770
CMS02780
```

```
C
C      PLOT PROGRAM.
C
700 X = PTXF(X1,Y1)
    Y = PTYF(Y1)
    IF (LD) 702,701,702
701 CALL PLOT(X,Y,2)
    GO TO 703
702 CALL PLTIR(X,Y,0)
703 X1 = X2
    Y1 = Y2
    GO TO LRET,(308,310,329)

C
C      329 INDEX1 = XNDEXF(IX-1,IY)
    IF (AM(INDEX1) - CV) 331,332,332
331 CALL PLACE(XNDEXF(IX,IY))
332 IS = IS + 5
    IX = IX1
    IY = IY1
    GO TO 333
334 INDEX1 = XNDEXF(IX-1,IY)
    IF (AM(INDEX1) - CV) 335,310,310
335 CALL PLACE(XNDEXF(IX,IY))
310 X = PTXF(X2,Y2)
    Y = PTYF(Y2)
    IF (LD) 337,336,337
336 CALL PLOT(X,Y,2)
    GO TO 338
337 CALL PLTIR(X,Y,0)
338 GO TO NRET,(105,110,116,122,128)

C
C      128 IF (I - MT1) 132,133,132
132 I = I + 1
    GO TO 129
133 IF (J - NT1) 134,136,134
134 J = J + 1
    GO TO 135
136 RETURN
    END(1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0)
```

```
CMS02790
CMS02800
CMS02810
CMS02820
CMS02830
CMS02840
CMS02850
CMS02860
CMS02870
CMS02880
CMS02890
CMS02900
CMS02910
CMS02920
CMS02930
CMS02940
CMS02950
CMS02960
CMS02970
CMS02980
CMS02990
CMS03000
CMS03010
CMS03020
CMS03030
CMS03040
CMS03050
CMS03060
CMS03070
CMS03080
CMS03090
CMS03100
CMS03110
CMS03120
CMS03130
CMS03140
CMS03150
CMS03160
CMS03170
CMS03180
```

```

*
*****
* CALLING SEQUENCE
* -----
* 1. FORTRAN = CALL CLEAN
*              CALL PLACE(I)
*              INALF(I)
* 2. FAP      = TSX      $CLEAN,4      TSX      $PLACE,4
*              TSX      $INAL,4      PZE      I
*              I = ARGUMENT DANS AC.
*              RESULTAT DANS AC.
*
* CETTE SUBROUTINE EST UTILISEE PAR -COMAP- AFIN DE MEMORISER LES
* COORDONNEES DES POINTS TRAITES DANS LA MATRICE -AM- -CLEAN-
* REMET A ZERO LA TABLE DE MEMORISATION, -PLACE- INDIQUE DANS LA
* TABLE DE MEMORISATION QUE LE POINT DE COORDONNEES -I- DE LA
* MATRICE -AM- EST TRAITÉ, ET -INAL- TESTE SI LE POINT DE COOR-
* DONNEES -I- DE LA MATRICE -AM- EST DÉJÀ TRAITÉ.
*
*****
* ( J. PIRE - 1/11/64 )
*
* C.C.R. EURATOM
* (ISPRA)

```

BINARY CARD NO. EUCAP000
 00000
 00005
 00030

ENTRY CLEAN
 ENTRY PLACE
 ENTRY INAL

CAP00290
 CAP00300
 CAP00310
 CAP00320

BINARY CARD NO. EUCAP001
 00000 0634 00 1 00040
 00001 0774 00 1 00426
 00002 0600 00 1 00474
 00003 2 00001 1 00002
 00004 0020 00 0 00040
 00005 -0625 00 0 00043
 00006 0634 00 1 00026
 00007 0500 60 4 00001
 00010 0402 00 0 00042
 00011 0771 00 0 00022
 00012 0131 00 0 00000
 00013 -0754 00 0 00000
 00014 0221 00 0 00045
 00015 0621 00 0 00024
 00016 0621 00 0 00034
 00017 0131 00 0 00000
 00020 0737 00 1 00000
 00021 -0520 00 0 00043
 00022 0020 00 0 00033
 00023 -0500 00 0 00044

CLEAN SXA XR1,1
 AXT 278,1
 STZ REC+278,1
 TIX *-1,1,1
 TRA XR1
 *
 PLACE STL SWITCH
 SXA XR11,1
 CLA* 1,4
 S1 SUB ONED
 ARS 13
 XCA
 ZAC
 DVP C36
 STA SHIFT1
 STA SHIFT2
 XCA
 PAC 0,1
 NZT SWITCH
 TRA S2
 CAL FIRST

CAP00330
 CAP00340
 CAP00350
 CAP00360
 CAP00370
 CAP00380
 CAP00390
 CAP00400
 CAP00410
 CAP00420
 CAP00430
 CAP00440
 CAP00450
 CAP00460
 CAP00470
 CAP00480
 CAP00490
 CAP00500
 CAP00510
 CAP00520
 CAP00530

EUCAP

*** 7090 SUBROUTINES FOR CALCOMP 564/1401. ***
 - SUBROUTINES CLEAN, PLACE, /FUNCTION INAL. -

11/11/64

PAGE 2

BINARY CARD NO. EUCAP002

00024	0771	00	0	00000	SHIFT1	ARS	**
00025	-0602	00	1	00046		ORS	REC, 1
00026	0774	00	1	00000	XR11	AXT	**, 1
00027	0020	00	4	00002		TRA	2, 4
*							
00030	0600	00	0	00043	INAL	STZ	SWITCH
00031	0634	00	1	00040		SXA	XR1, 1
00032	0020	00	0	00010		TRA	S1
00033	0560	00	1	00046	S2	LDQ	REC, 1
00034	-0773	00	0	00000	SHIFT2	RQL	**
00035	0500	00	0	00042		CLA	ONED
00036	0162	00	0	00040		TQP	**+2
00037	-0754	00	0	00000		ZAC	
00040	0774	00	1	00000	XR1	AXT	**, 1
00041	0020	00	4	00001		TRA	1, 4
*							
00042	0	00001	0	00000	ONED	PZE	0, 1
00043	0	00000	0	00000	SWITCH	PZE	
00044	-0	00000	0	00000	FIRST	MZE	
00045	0	00000	0	00044	C36	PZE	36
*							
00046					REC	BSS	278
*							
*							
END							

CAP00540
 CAP00550
 CAP00560
 CAP00570
 CAP00580
 CAP00590
 CAP00600
 CAP00610
 CAP00620
 CAP00630
 CAP00640
 CAP00650
 CAP00660
 CAP00670
 CAP00680
 CAP00690
 CAP00700
 CAP00710
 CAP00720
 CAP00730
 CAP00740
 CAP00750
 CAP00760
 CAP00770
 CAP00780


```

C      (7090 SUBROUTINE FOR CALCOMP 564/1401)
C*****
C* CALLING SEQUENCE
C* 1. FORTRAN = CALL AXMAP(NXL,NYL,DELTAX,DELTAY,THETA,BCDX,NOX,
C*                   BCDY,NOY)
C* 2. FAP      = TSX      $AXMAP,4
C*               PZE      NXL
C*               PZE      NYL
C*               .
C*               PZE      NOY
C*
C* NXL      = DIMENSION UTILISEE DES X DANS LA MATRICE(VOIR COMAP).
C* NYL      = DIMENSION UTILISEE DES Y DANS LA MATRICE(VOIR COMAP).
C* DELTAX   = INTERVALLE DE DESSIN POUR LES X.
C* DELTAY   = INTERVALLE DE DESSIN POUR LES Y.
C* THETA    = ANGLE EN DEGRES(AXE Y).
C* BCDX     = VECTEUR CONTENANT LE TITRE POUR L AXE DES X.
C* NOX      = NOMBRE DE CARACTERES DU TITRE X.
C* BCDY     = VECTEUR CONTENANT LE TITRE POUR L AXE DES Y.
C* NOY      = NOMBRE DE CARACTERES DU TITRE Y.
C*****
C      ( J. PIRE - 1/11/64 )
C
C      SUBROUTINE AXMAP(NXL,NYL,DELTAX,DELTAY,THETA,BCDX,NOX,BCDY,NOY)
C
C      NXLG = NXL + 1
C      NYLG = NYL + 1
C      PASX = DELTAX
C      PASY = DELTAY
C      THE = THETA
C      XN = FLOATF(NOX)
C      YN = FLOATF(NOY)
C      SIGNX = PASX / ABSF(PASX)
C      SIGNY = PASY / ABSF(PASY)
C      THX = 180. * 0.017453294
C      THY = THE * 0.017453294
C
C      TRACE AXIS.
C
C      CALL PLOT (0.,0.,3)
C      X1 = FLOATF(NXLG) * PASX
C      CALL PLOT (X1,0.,2)
C      STHY = SIN(THY)
C      SIGNA = STHY / ABSF(STHY)
C      CTHY = COS(THY)
C      SIZEY = FLOATF(NYLG) * PASY
C      X3 = SIZEY * CTHY

```

```

AMS00010
AMS00020
AMS00030
AMS00040
AMS00050
AMS00060
AMS00070
AMS00080
AMS00090
AMS00100
AMS00110
AMS00120
AMS00130
AMS00140
AMS00150
AMS00160
AMS00170
AMS00180
AMS00190
AMS00200
AMS00210
AMS00220
AMS00230
AMS00240
AMS00250
AMS00260
AMS00270
AMS00280
AMS00290
AMS00300
AMS00310
AMS00320
AMS00330
AMS00340
AMS00350
AMS00360
AMS00370
AMS00380
AMS00390
AMS00400
AMS00410
AMS00420
AMS00430
AMS00440
AMS00450
AMS00460
AMS00470
AMS00480
AMS00490
AMS00500
AMS00510
AMS00515
AMS00520
AMS00530
AMS00540

```

EUAMS

- AXIS MAP SUBROUTINE. -

11/11/64

```

      X2 = X1 + X3
      Y = SIZEY * STHY
      CALL PLOT (X2,Y,2)
      CALL PLOT (X3,Y,2)
      CALL PLOT (0.,0.,2)
C
C      WRITE X GRADUATIONS.
C
      TH1 = 0.
      TH2 = THY
      DL = PASX
      XA = 0.
      YA = 0.
      N = NXLG - 1
      VAL = 1.
      ANGLE = THE + 90. + 90.*SIGNX
      VD = 1.
      SIGN1 = SIGNX
      SIGN2 = SIGNY
      ASSIGN 100 TO IRET
      GO TO 500
100  XA = X1/2. - 0.09*XN
      YA = SIGNA*SIGNY*0.40 - 0.11
      CALL SYMBL4 (XA,YA,0.21,0.,BCDX,NOX)
C
C      MARK Y AXIS.
C
      TH1 = THY
      TH2 = THX
      DL = PASY
      XA = X1
      YA = 0.
      N = NYLG - 1
      VAL = -1.
      SIGN1 = SIGNY
      SIGN2 = SIGNX
      ASSIGN 110 TO IRET
      GO TO 500
C
C      MARK X AXIS.
C
110  TH1 = THX
      TH2 = THX + THY
      DL = PASX
      XA = X2
      YA = Y
      N = NXLC - 1
      SIGN1 = SIGNX
      SIGN2 = SIGNY
      ASSIGN 120 TO IRET
      GO TO 500
C
C      WRITE Y GRADUATIONS.
C
120  TH1 = THX + THY
      TH2 = 0.

```

```

AMS00550
AMS00560
AMS00570
AMS00580
AMS00590
AMS00600
AMS00610
AMS00620
AMS00630
AMS00640
AMS00650
AMS00660
AMS00670
AMS00680
AMS00690
AMS00700
AMS00710
AMS00720
AMS00730
AMS00740
AMS00750
AMS00760
AMS00770
AMS00780
AMS00790
AMS00800
AMS00810
AMS00820
AMS00830
AMS00840
AMS00850
AMS00860
AMS00870
AMS00880
AMS00890
AMSC0900
AMSC0910
AMS00920
AMS00930
AMS00940
AMS00950
AMS00960
AMS00970
AMS00980
AMSC0990
AMS01000
AMS01010
AMS01020
AMS01030
AMS01040
AMS01050
AMS01060
AMS01070
AMS01080
AMS01090
AMS01100

```

```

DL = PASY
XA = X3
YA = Y
N = NYLG - 1
VAL = N
ANGLE = 0.
VD = -1.
SIGN1 = -1.
SIGN2 = SIGNX
ASSIGN 130 TO IRET

C
C
C EXECUTE ROUTINE.
500 STH1L = DL * SINP(TH1)
    CTH1L = DL * COSP(TH1)
    STH2 = SINP(TH2)
    CTH2 = COSP(TH2)
    XI = 0.10 * CTH2 * SIGN2
    YI = 0.10 * STH2 * SIGN2
    FIX1 = 0.07 * STH2 * SIGN1 + XI
    FIX2 = 0.07 * CTH2 * SIGN1 - YI
    DO 501 I = 1, N
        XA = XA + CTH1L
        YA = YA + STH1L
        CALL PLOT (XA, YA, 3)
        XB = XA + XI
        YB = YA + YI
        CALL PLOT (XB, YB, 2)
        CALL PLOT (XA, YA, 2)
        IF (VAL) 501, 502, 502
502 IF (VAL - 10.) 504, 505, 503
503 IF (VAL - 99.) 505, 505, 506
504 CN = 0.06
    GO TO 507
505 CN = 0.12
    GO TO 507
506 CN = 0.18
507 XB = XA - FIX1 + (SIGN1 - SIGN2)*CN*CTH2
    YB = YA + FIX2 + (SIGN1 - SIGN2)*CN*STH2
    CALL NUMBER (XB, YB, 0.14, ANGLE, VAL, -1)
    VAL = VAL + VD
501 CONTINUE
    GO TO IRET, (100, 110, 120, 130)

C
C
130 FIX1 = SIZEY/2. - 0.09*YN
    FIX2 = SIGNA*SIGNX*0.40 + 0.11
    XA = FIX1*CTHY + FIX2*STHY
    YA = FIX1*STHY - FIX2*CTHY
    CALL SYMBL4 (XA, YA, 0.21, THE, BCDY, NOY)
900 RETURN
    END(1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0)

```

```

AMSO1110
AMSO1120
AMSO1130
AMSO1140
AMSO1150
AMSO1160
AMSO1170
AMSO1180
AMSO1190
AMSO1200
AMSO1210
AMSO1220
AMSO1230
AMSO1240
AMSO1250
AMSO1260
AMSO1270
AMSO1280
AMSO1290
AMSO1300
AMSO1310
AMSO1320
AMSO1330
AMSO1340
AMSO1350
AMSO1360
AMSO1370
AMSO1380
AMSO1390
AMSO1400
AMSO1410
AMSO1420
AMSO1430
AMSO1440
AMSO1450
AMSO1460
AMSO1470
AMSO1480
AMSO1490
AMSO1500
AMSO1510
AMSO1520
AMSO1530
AMSO1540
AMSO1550
AMSO1560
AMSO1570
AMSO1580
AMSO1590
AMSO1600
AMSO1610

```

```

C      DIMENSION AM(50,50)
C
100  DO 1 J = 1,25
      READ INPUT TAPE 5,100,(AM(I,J),I=1,19)
      FORMAT (19(1X,F3.1))
      CALL FINIM (0.,4.)
      CALL PAGE
      CALL SYMBL4 (4.5,27.,0.5,0.,17HEXEMPLE NUMERO 1.,17)
      CALL FINIM (4.5,2.5)
      CALL COMAP (AM,19,25,50,50,0.5,0.9,3.5,80.,0)
      CALL COMAP (AM,19,25,50,50,0.5,0.9,6.5,80.,1)
      CALL AXMAP (19,25,0.5,0.9,80.,6HDIM. X,6,6HDIM. Y,6)
      CALL FINIM (-4.5,32.)
      CALL PAGE
      CALL SYMBL4 (4.5,27.,0.5,0.,17HEXEMPLE NUMERO 2.,17)
      CALL FINIM (8.5,25.5)
      CALL COMAP (AM,19,25,50,50,0.5,-0.9,3.5,80.,0)
      CALL COMAP (AM,19,25,50,50,0.5,-0.9,6.5,80.,1)
      CALL AXMAP (19,25,0.5,-0.9,80.,6HDIM. X,6,6HDIM. Y,6)
      CALL FINIM (17.5,-60.)
      CALL PAGE
      CALL SYMBL4 (4.5,27.,0.5,0.,17HEXEMPLE NUMERO 3.,17)
      CALL FINIM (18.5,25.5)
      CALL COMAP (AM,19,25,50,50,-0.5,-0.9,3.5,80.,0)
      CALL COMAP (AM,19,25,50,50,-0.5,-0.9,6.5,80.,1)
      CALL AXMAP (19,25,-0.5,-0.9,80.,6HDIM. X,6,6HDIM. Y,6)
      CALL FINIM (-18.5,9.)
      CALL PAGE
      CALL SYMBL4 (4.5,27.,0.5,0.,17HEXEMPLE NUMERO 4.,17)
      CALL FINIM (14.5,2.5)
      CALL COMAP (AM,19,25,50,50,-0.5,0.9,3.5,80.,0)
      CALL COMAP (AM,19,25,50,50,-0.5,0.9,6.5,80.,1)
      CALL AXMAP (19,25,-0.5,0.9,80.,6HDIM. X,6,6HDIM. Y,6)
      CALL FINIM (11.5,-41.)
      CALL FINTRA
      WRITE OUTPUT TAPE 6,101
101  FORMAT (1H1,5X,7HMATRIX.,//)
      DO 2 J = 1,25
102  WRITE OUTPUT TAPE 6,102,(AM(I,J),I=1,19)
      FORMAT (1H0,5X,19(F3.1,2X))
      CALL EXIT
      END(1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0)

```

```

MAIN0010
MAIN0020
MAIN0030
MAIN0040
MAIN0050
MAIN0060
MAIN0070
MAIN0080
MAIN0090
MAIN0100
MAIN0110
MAIN0120
MAIN0130
MAIN0140
MAIN0150
MAIN0160
MAIN0170
MAIN0180
MAIN0190
MAIN0200
MAIN0210
MAIN0220
MAIN0230
MAIN0240
MAIN0250
MAIN0260
MAIN0270
MAIN0280
MAIN0290
MAIN0300
MAIN0310
MAIN0320
MAIN0330
MAIN0340
MAIN0350
MAIN0360
MAIN0370
MAIN0380
MAIN0390
MAIN0400
MAIN0410
MAIN0420

```

PAGE - TRACE DE LA PAGE. -

11/11/64

PAGE 1

C

SUBROUTINE PAGE

C

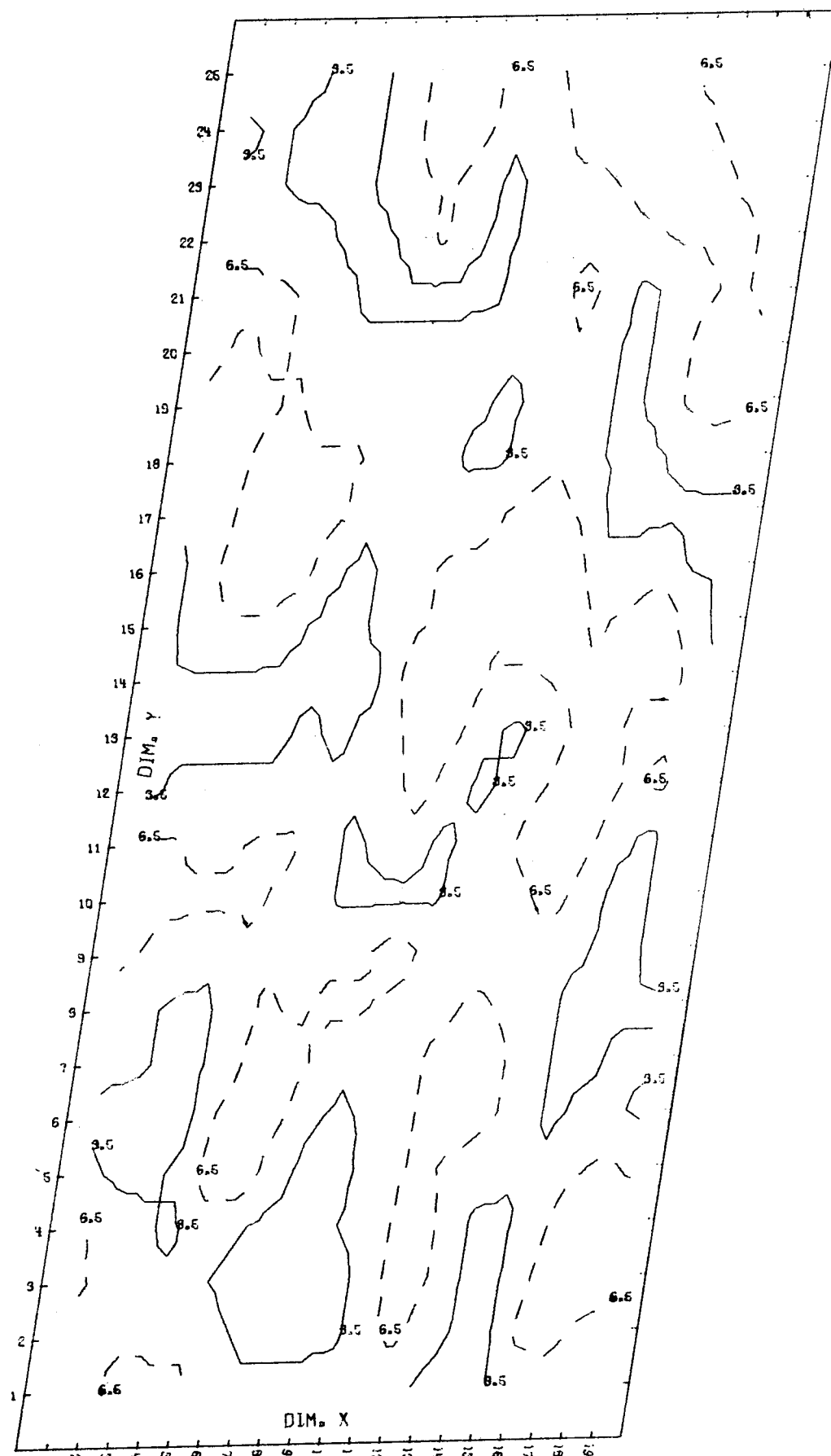
CALL PLTIR (21.,0.,2)
CALL PLTIR (21.,29.5,2)
CALL PLTIR (0.,29.5,2)
CALL PLTIR (0.,0.,2)
CALL PLOT (3.5,1.,3)
CALL PLOT (20.,1.,2)
CALL PLOT (20.,28.5,2)
CALL PLOT (3.5,28.5,2)
CALL PLOT (3.5,1.,2)
RETURN
END(1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0)

PAGE0010
PAGE0020
PAGE0030
PAGE0040
PAGE0050
PAGE0060
PAGE0070
PAGE0080
PAGE0090
PAGE0100
PAGE0110
PAGE0120
PAGE0130

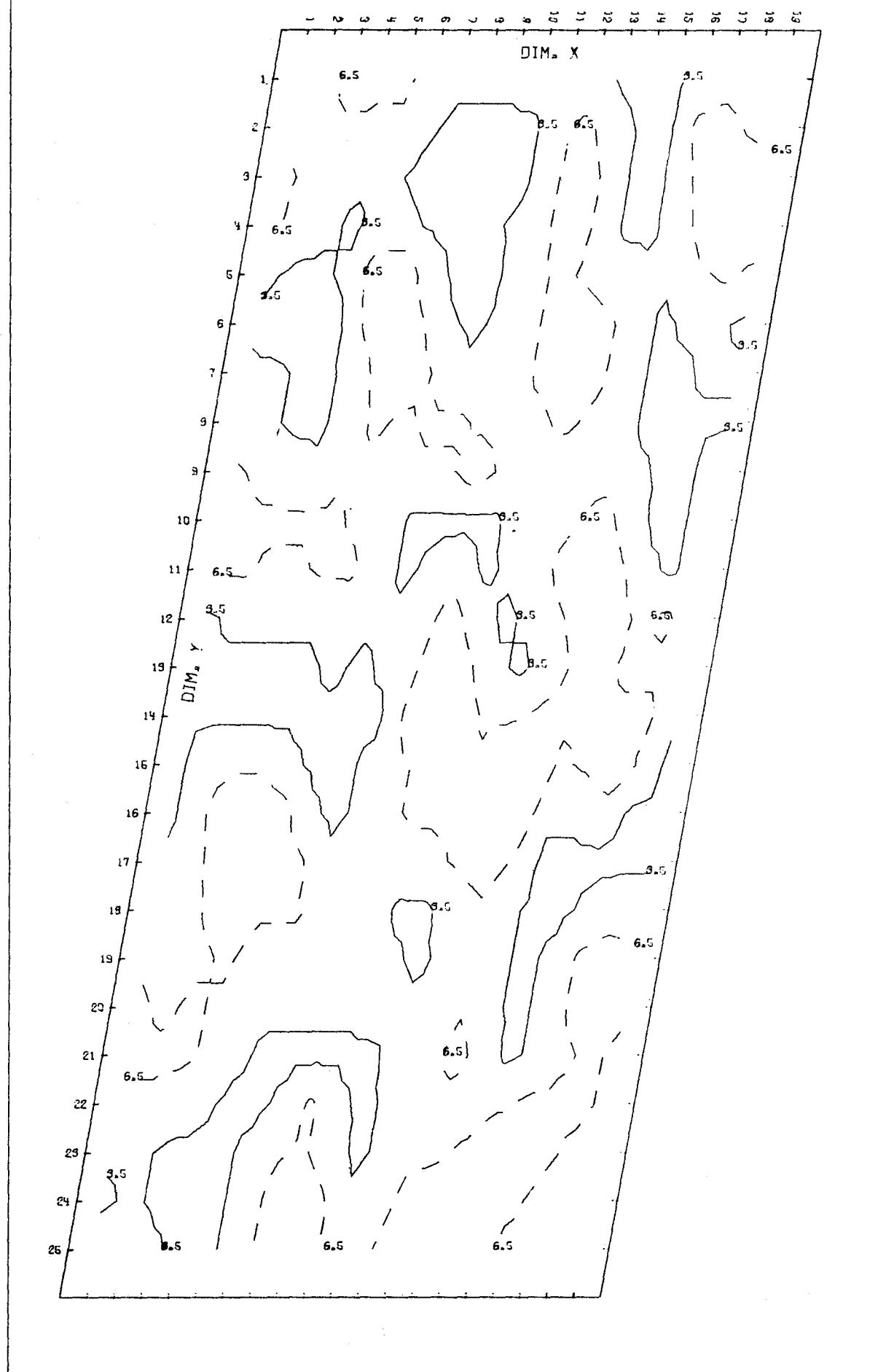
MATRIX.

5.0	5.0	8.0	7.0	7.0	5.0	4.0	4.0	4.0	5.0	6.0	5.0	3.0	2.0	3.0	5.0	5.0	4.0	4.0
5.0	6.0	6.0	6.0	6.0	4.0	3.0	3.0	3.0	3.0	6.0	7.0	5.0	3.0	3.0	7.0	8.0	6.0	5.0
7.0	6.0	4.0	4.0	4.0	3.0	2.0	2.0	2.0	3.0	6.0	8.0	6.0	3.0	3.0	6.0	9.0	9.0	8.0
7.0	5.0	4.0	3.0	5.0	4.0	3.0	3.0	3.0	4.0	6.0	8.0	6.0	3.0	3.0	5.0	8.0	8.0	8.0
4.0	3.0	3.0	4.0	8.0	9.0	5.0	3.0	2.0	4.0	6.0	7.0	6.0	5.0	4.0	4.0	6.0	7.0	6.0
3.0	2.0	2.0	3.0	7.0	9.0	7.0	4.0	3.0	4.0	6.0	7.0	8.0	7.0	5.0	3.0	4.0	4.0	3.0
4.0	4.0	3.0	3.0	5.0	8.0	8.0	6.0	4.0	5.0	6.0	7.0	8.0	7.0	5.0	3.0	3.0	4.0	4.0
5.0	4.0	3.0	3.0	5.0	7.0	6.0	7.0	7.0	6.0	6.0	6.0	7.0	6.0	5.0	3.0	3.0	3.0	3.0
7.0	6.0	5.0	4.0	6.0	6.0	5.0	6.0	6.0	8.0	6.0	6.0	5.0	5.0	5.0	4.0	3.0	3.0	6.0
8.0	7.0	7.0	7.0	7.0	6.0	5.0	3.0	3.0	3.0	3.0	5.0	5.0	6.0	8.0	5.0	3.0	3.0	6.0
7.0	7.0	6.0	6.0	7.0	7.0	5.0	3.0	4.0	5.0	3.0	4.0	5.0	8.0	9.9	7.0	4.0	3.0	6.0
3.0	4.0	4.0	4.0	4.0	5.0	4.0	4.0	5.0	8.0	5.0	3.0	4.0	6.0	8.0	8.0	6.0	7.0	5.0
3.0	3.0	3.0	3.0	3.0	4.0	3.0	4.0	6.0	9.0	7.0	4.0	3.0	5.0	7.0	7.0	6.0	6.0	5.0
3.0	3.0	3.0	3.0	3.0	3.0	3.0	3.0	7.0	9.9	8.0	6.0	6.0	7.0	7.0	7.0	7.0	7.0	4.0
3.0	6.0	6.0	6.0	4.0	3.0	3.0	4.0	6.0	7.0	7.0	7.0	9.9	9.9	6.0	7.0	9.0	6.0	3.0
3.0	6.0	8.0	9.0	8.0	4.0	3.0	4.0	6.0	7.0	7.0	7.0	9.9	8.0	4.0	4.0	5.0	3.0	3.0
4.0	5.0	8.0	9.9	8.0	7.0	4.0	5.0	5.0	5.0	6.0	7.0	8.0	6.0	3.0	3.0	3.0	3.0	3.0
5.0	5.0	7.0	8.0	7.0	7.0	6.0	5.0	5.0	3.0	3.0	5.0	6.0	4.0	3.0	3.0	4.0	5.0	5.0
6.0	5.0	6.0	7.0	5.0	5.0	6.0	5.0	5.0	4.0	3.0	4.0	5.0	4.0	3.0	4.0	7.0	8.0	7.0
7.0	6.0	7.0	6.0	4.0	4.0	4.0	4.0	4.0	5.0	4.0	4.0	6.0	4.0	3.0	4.0	7.0	8.0	7.0
7.0	7.0	7.0	6.0	4.0	3.0	3.0	3.0	3.0	3.0	5.0	4.0	8.0	5.0	3.0	4.0	6.0	7.0	6.0
6.0	6.0	5.0	4.0	3.0	3.0	4.0	7.0	4.0	3.0	4.0	5.0	5.0	5.0	6.0	7.0	7.0	7.0	6.0
4.0	4.0	3.0	3.0	3.0	4.0	6.0	7.0	5.0	3.0	4.0	6.0	6.0	7.0	9.0	8.0	7.0	6.0	5.0
3.0	4.0	3.0	3.0	3.0	4.0	7.0	9.0	7.0	4.0	5.0	7.0	8.0	9.0	9.0	7.0	6.0	5.0	4.0
5.0	6.0	4.0	3.0	3.0	4.0	7.0	8.0	7.0	6.0	6.0	8.0	8.0	8.0	8.0	6.0	6.0	5.0	4.0

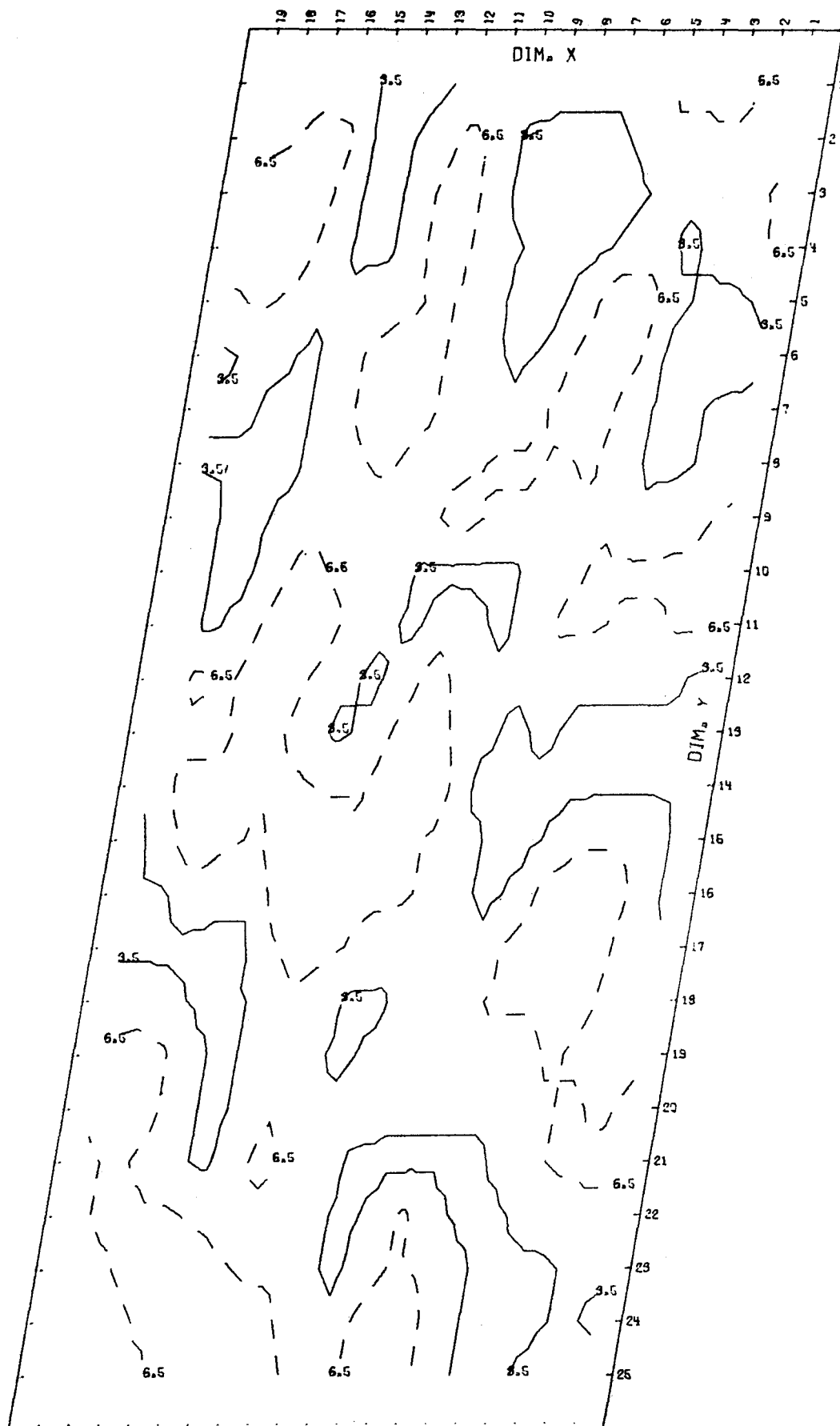
EXEMPLE NUMERO 1.



EXEMPLE NUMERO 2.



EXAMPLE NUMBER 3.



EXEMPLE NUMERO 4.

